SeqFIRE

User Manual version 1.1

January 2014

Pravech Ajawatanawong
Systematic Biology, Evolutionary Biology Centre (EBC)
Uppsala University, Sweden
January 2014

© Copyright 2012-2014 by SeqFIRE Development Team.
SeqFIRE web application, standalone version, and this documentation, are distributed free of charge for all use.

Preface

This manual is for the general user who wishes to use SeqFIRE online or standalone. The manual is divided into 6 chapters. If you want to use SeqFIRE quickly, using default settings, you can go directly to Chapter 1. For users who want to adjust some parameters for the analysis, it is more useful to understand the algorithms behind the program. The details of these algorithms are provided in Chapters 2 and 3.

For the analysis of multiple data sets or high-throughput analysis, SeqFIRE also provides a batch mode. The details for using the batch mode are provided in Chapter 4. For the advanced user who wants to use standalone SeqFIRE or pipeline SeqFIRE, this information is in Chapter 5. The algorithms in Chapters 2 and 3 are also useful for these users.

Finally Chapter 6 has suggestions for error messages. If you find any errors or have any questions or comments, please send these to me at pravech.a@gmail.com.

I would like to thank you Prof. Sandra L. Baldauf and Dr. Allison Perrigo for the great comments and proof reading this manual.

Pravech Ajawatanawong
January 2014

Contents

Preface i
Contents ii

Chapter 1 Get Started with SeqFIRE 1

What is SeqFIRE? 1
Input File 2
The Indel Region Module 2
Output from the Indel Region Module 4
The Conserved Block Module 5
Output from the Conserved Block Module 5

Chapter 2 Indel Region Module 7

Structure of an Alignment 7
Indel Region Module Algorithm 8
Generation of the Gap Profile 8
Partial Treatment 9
Generation of the Conservation Profile 10
Twilight Treatment 11
Generation of the Indel Profile 11
Output Page 12
Alignment with Indel Profile in Jalview Visualization 13
Alignment with Indel Annotation 13
Indel List 14
Indel Matrix 15
Masked Indel Alignment (Alignment without Indels) 16

Chapter 3 Conserved Block Module 17

Similarity and Information Entropy 17
Conserved Block Module Algorithm 18
Generation of the Gap Profile 19
Conserved Block Identification Using Similarity Scoring 19
Conserved Block Identification Using Entropy Scoring 20
Combining the Conserved Blocks from the Two Scoring Techniques 20
Output Page 20
Co-analysis between Conserved Block and Indel Region Modules 21

Chapter 4 Working with Multiple Datasets 22

Installation of SeqFIREprep 22
Preparation of the Input Data 22
Using the Batch Mode 23
Separation of the Outputs 24

Chapter 5 Running SeqFIRE Locally 26

Installation 26
General Options 26
Help Option (-h) 26
Input Option (-i) 26
Analysis Mode Option (-a) 27
Output Option (-o) 27
Indel Region Module Options 27
Similarity Threshold Option (-c) 27
Substitution Group Option (-g) 28
Inter-indel Space Option (-b) 28
Partial Treatment Option (-p) 28
Twilight Treatment Option (-t) 29
Options for Conserved Block Module 29
Percent Accept Gap Option (-j) 29
Similarity Threshold Option (-d) 30
Substitution Group Option (-k) 30
Minimum Space between Two Blocks Option (-s) 30
Maximum Size for Non-conserved Block Option (-f) 31
Conserved Block Combination Option (-r) 31
Options for SpecialAnalyses 31
Co-analysis (Indel Region & Conserved Block) Option (-e) 31
Multiple Data Analysis Option (-m) 32
SeqFIRE Quick Run 32
Running the Indel Region Module 32
Running the Conserved Block Module 32

Chapter 6 Error Messages 33

Error Messages 33
Parameter Value out of Range 33
Input Conflict 33
No Input 33
Input Cannot Run 34

References 35

Chapter 1

Getting Start with SeqFIRE

What is SeqFIRE?

SeqFIRE is a user-friendly web application for the identification and extraction of indel and conserved blocks from multiple sequence alignments. The output is provided in several different formats, which can be useful as input for further analyses, such as phylogenetic analysis. Users do not need to install any prerequisite software in order to use SeqFIRE. It can be accessed online at the URL: www.seqfire.org/ (Ajawatanawong *et al.*, 2012). The SeqFIRE main page is shown in Figure 1.1.

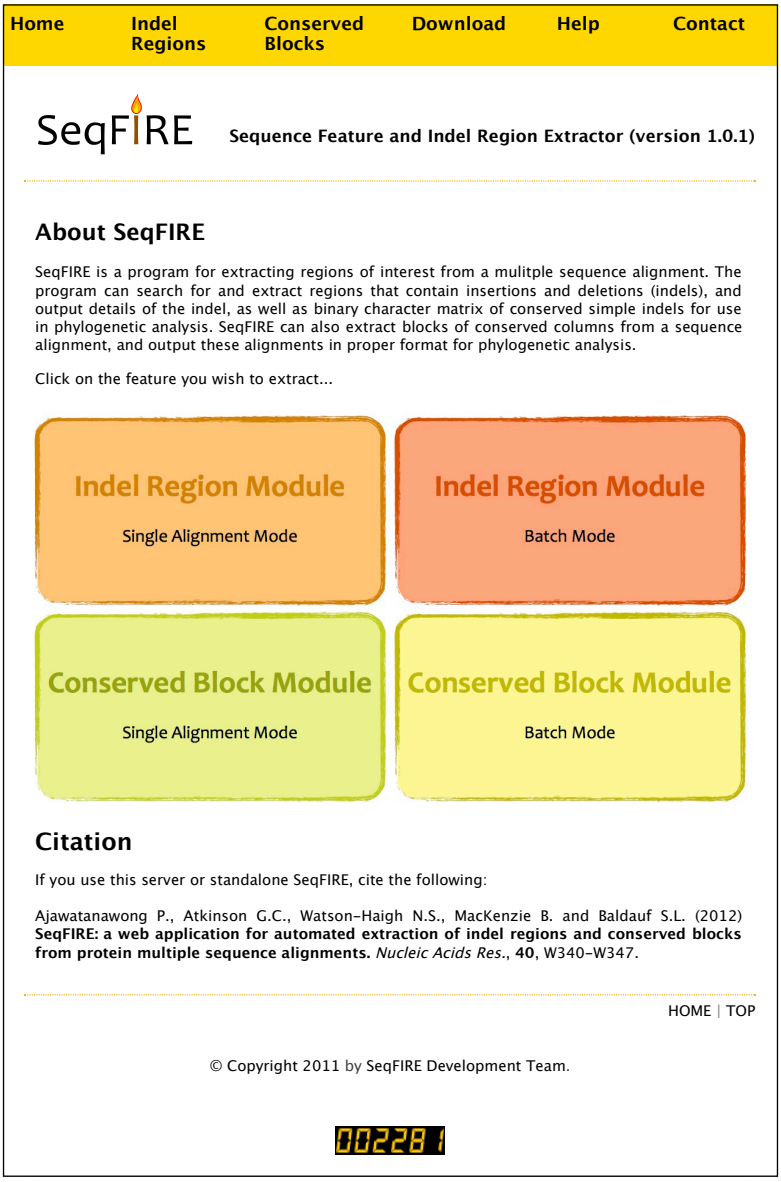


Figure 1.1 Home page of SeqFIRE (www.seqfire.org/).

The program comprises six tabs as follows:

- **Home:** home page of the program
- **Indel Regions:** input page for the indel module
- **Conserved Blocks:** input page for the conserved block module
- **Download:** link for downloading SeqFIREprep, a standalone version for multiple data analysis (multiple inputs)
- **Help:** links for the online help Wiki, the manual in PDF format and some useful FAQs
- **Contact:** credits and e-mails of the developers

Input File

SeqFIRE consists of two different modules: an **Indel Region Module** and a **Conserved Block Module**. Both modules require a protein alignment in FastA format as input, so, if you have unaligned protein sequences, you will have to align them first. On the Help tab, we provide links to some useful alignment program, such as **MUSCLE** (www.drive5.com/muscle/) (Edgar, 2004a; Edgar, 2004b), **ProbCons** (probcons.stanford.edu/) (Do *et al.*, 2005) **MAFF-T** (http://mafft.cbrc.jp/alignment/software/) (Katoh & Standley, 2013), **webPRANK** (www.ebi.ac.uk/goldman-srv/prank/prank/) (Löytynoja & Goldman, 2010), and **K-align** (http://www.braembl.org.au/tools/kalign-multiple-sequence-alignment) (Lassmann & Sonnhammer, 2005). These are all easy to use advanced iterative alignment programs that give good quality alignments for even quite divergent sequences. Outputs from any other alignment program will also work in SeqFIRE as long as the output is in FastA format.

The Indel Region Module

The SeqFIRE indel region module identifies and extracts indel regions using default or user-defined criteria. These criteria are used to calculate a consensus sequence from your alignment. This consensus is then used to define indel regions.

Input: Sequences can be input either by copy/pasting them into the input box or by uploading the file using the Choose File button. If you want to run SeqFIRE with the default example alignment, just click the Load example alignment button, and the example protein alignment will appear in the input box.

Parameters: SeqFIRE uses five adjustable parameters to identify indel regions (Figure 1.2). The parameters are as follows:

- **Amino acid conservation threshold:** This is the percentage sequence similarity required for an alignment position to be included in the consensus. The default is 75% similarity.
- **Amino acid substitute group:** This selects the amino acid substitution model for scoring sequence similarity. SeqFIRE provides six alternative matrices: PAM60, PAM250, BLOSUM40, BLOSUM62, BLOSUM80, and NONE. The default setting is NONE, where all amino acid differences are weighted equally.
- **Inter-indel space:** This is the minimum number of consensus sites required between two indels, in order for them to be treated as separate indels. The default is three sites.
- **Detect partial sequence:** This allows the program to search for indels in the N- and C-terminal ends of the alignment, in cases where some sequences are incomplete (short sequences). The default is “no,” which does not allow searching in the terminal ends of the alignment.

- **Twilight treatment:** This allows indel searching in alignments with highly divergent sequences. This automatically sets the similarity score cut off to 30% to determine the conserved regions. This is based on the concept of "twilight zone proteins" where two different sequences may still have the same structure with even similarity as low as 30% (Rost, 1999). The default is "no".

Once all parameters are set, press the FIRE! button to begin the analysis.

Note: If no parameters are selected the program will execute using the default parameters.

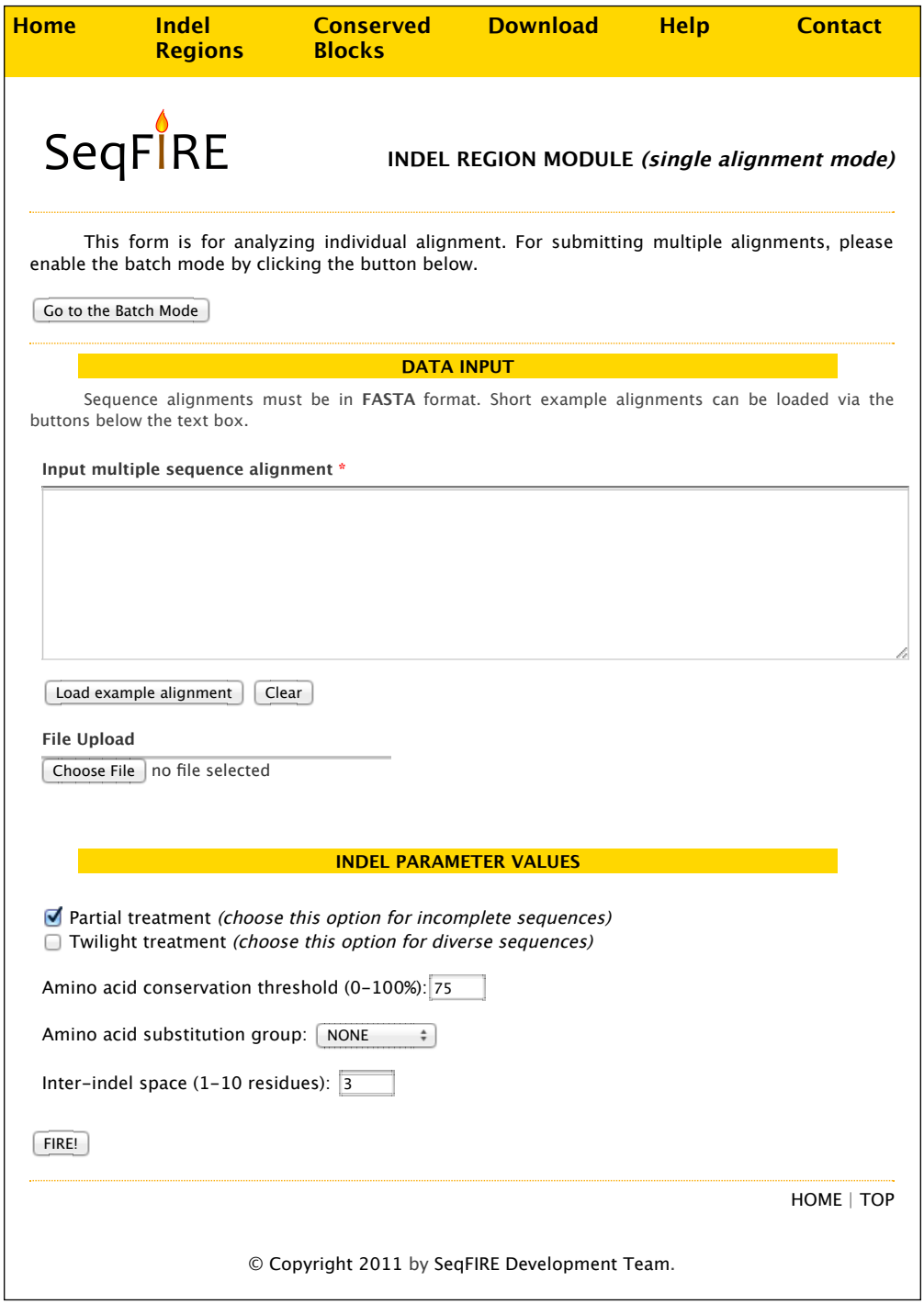


Figure 1.2 The Indel Region tab.
After uploading an input file (in FastA format), parameters are adjusted under Indel Parameter Values in the lower panel. Pressing the "FIRE" button executes the program.

Output from the Indel Region Module

SeqFIRE provides several output formats in this module. These are …

1) alignment with indel profile in Jalview (Waterhouse, *et al.*, 2009)
2) alignment with indel annotation
3) indel list
4) indel matrix in NEXUS format
5) alignment with indel region masked (see next chapter for details)

These outputs can be accessed using the links at the top of the output page (see Figure 1.3) o using links at the beginning of each section below.. Results can be downloaded for any or all of these results by right-clicking the link and saving the linked file.

The results page will also display your protein alignment with its indel profile below (the purple line underneath the alignment in Figure 1.3) and its conservation profile below that (the yellowish bars in Figure 1.3) in Jalview.

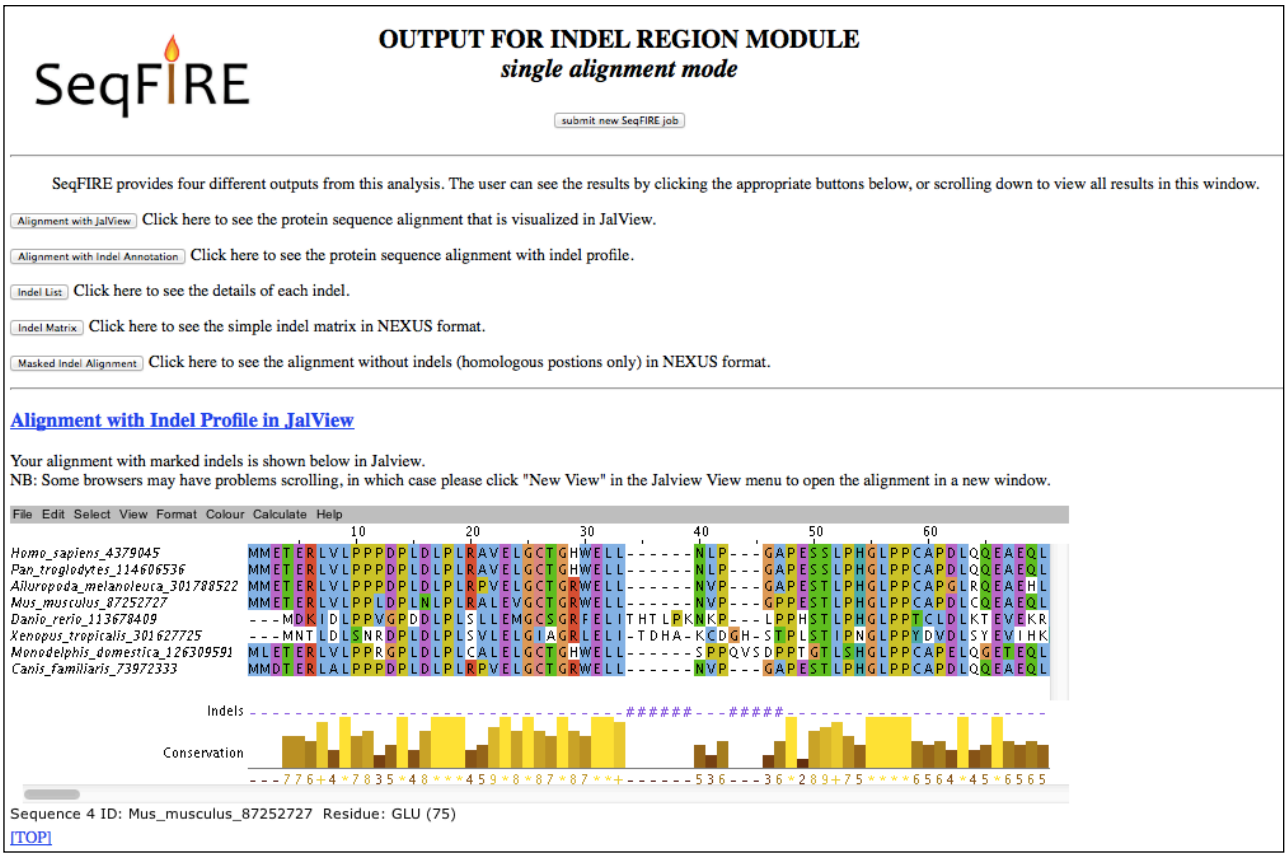


Figure 1.3 Top position of results page of SeqFIRE.
At the top of page, there are five links for downloading: (1) alignment in Jalview, (2) alignment with indel annotation, (3) indel list, (4) NEXUS format of the indel matrix (if the program detects simple indels in your alignment), and (5) alignment with indel regions masked. The program then displays all outputs sequentially, beginning with your protein alignment with the indel and conserved references in Jalview (Waterhouse, *et al.*, 2009).

The Conserved Block Module

In order to use a molecular sequence alignment for phylogenetic analysis, gaps >1-2 alignment positions and any flanking regions of low sequence conservation should be removed (Castresana, 2000; Löytynoja and Goldman, 2008; Wu, *et al.*, 2012). Incorporation of these data are likely to contribute only noise to phylogenetic analysis. This is because most phylogenetic reconstruction software implements substitution models only (not insertion/deletion models). So, as a result the user has to look at the alignment and decide to keep or discard uncertain alignment regions manually, which is tedious work and often lacks systematic criteria.

The purpose of the conserved region module is to identify and extract alignment regions of certain homology. The module is accessed by clicking the "Conserved block" link or the "Conserved blocks" tab at the top of the input page. This takes you to the Conserved blocks input page (Figure 1.4).

Similar to the indel region module, files in FastA format can be uploaded either by direct copy/ paste or via the Choose File button. There are also five adjustable parameters to help determine the conserved regions. The first two parameters are identical to the indel region module parameters.

- **Amino acid conservation threshold:** This is the percentage sequence similarity required for an alignment position to be included in the consensus. The default is 75% similarity.
- **Amino acid substitute group:** This selects the amino acid substitution model. SeqFIRE provides six alternative matrices: PAM60, PAM250, BLOSUM40, BLOSUM62, BLOSUM80, and NONE. The default setting is NONE, where all amino acid differences are weighted equally.
- **Minimum size of conserved block:** This is the minimum number of adjacent positions in the alignment used as a criterion for the program to determine a conserved block. The default is three sites.
- **Maximum size of non-conserved block:** This is the maximum number of poorly conserved (absent from consensus sequence) alignment positions that are allowed to be part of a conserved block. The default is three sites.
- **Maximum percentage of gaps allowed in a conserved column:** Some gapped positions in the alignment might still include informative sites for phylogenetic reconstruction. This criterion is a cut off to tell the program what percentage of gaps (ratio of gaps per column) should be retained or discarded in gapped positions in the conserved block. The default is 40%.
- **Combination of conserved block profiles:** SeqFIRE computes conserved blocks using two different methods, similarity and entropy. The resulting profiles can be combined either as union (all alignment columns in either similarity or entropy profiles) or intersection (only alignment in both similarity and entropy profiles). The default is "Union".
- **Co-analysis with indel region module:** Users can run both the indel regions and conserved region analyzes simultaneously by selecting "co-analysis with indel region module". The default is "Use conserved block module alone".

Output from the Conserved Block Module

The general result page of the conserved region of SeqFIRE is similar to the indel region output page. On top of the page, you will get a number of links for results in several

formats (see chapter 3 for details). Below this, the alignment with conserved block profile will be displayed in Jalview.

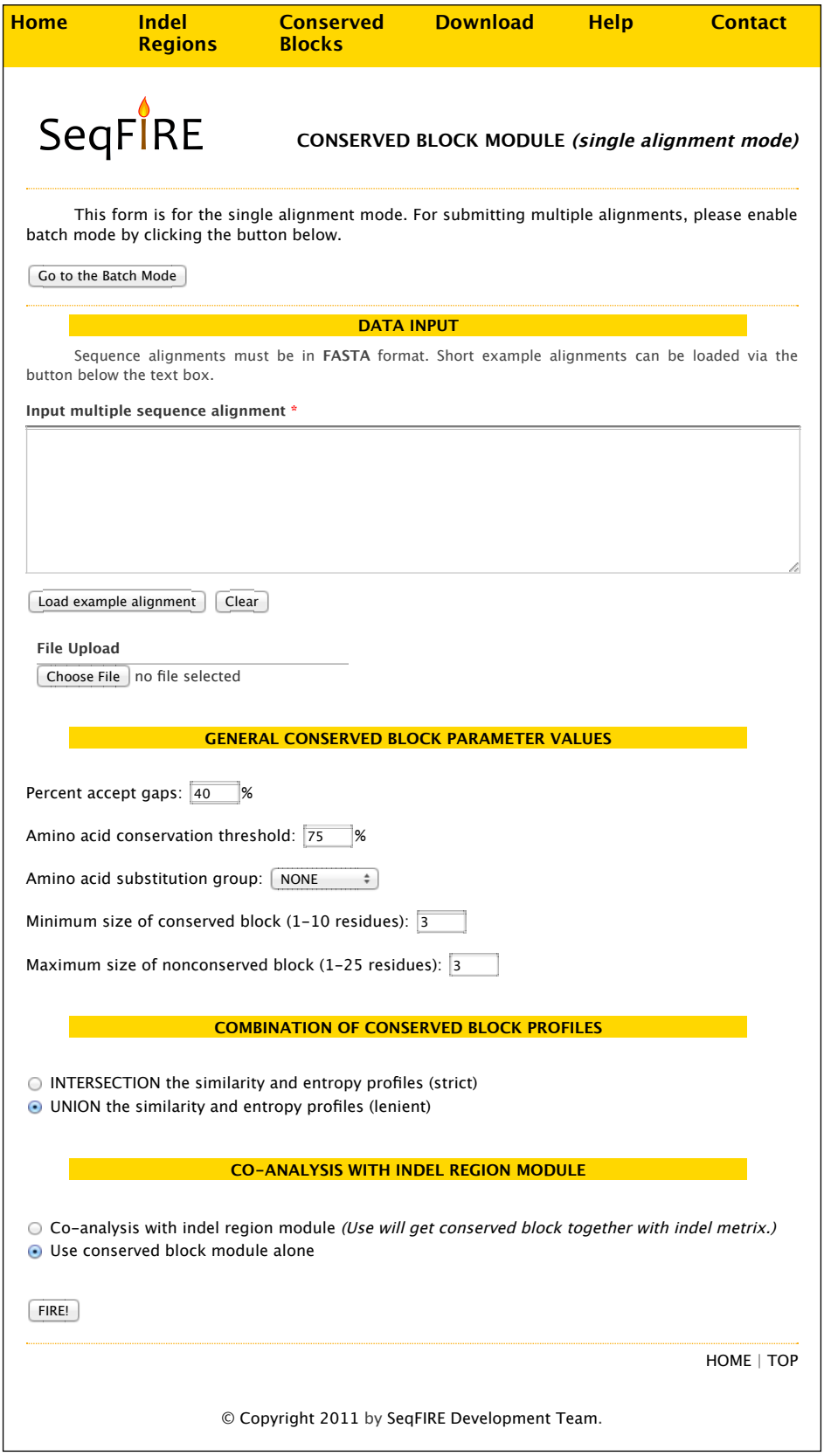


Figure 1.4 The conserved blocks module page.
The upper part of the page is the input section. The lower part is the section for setting parameters (see text).

Chapter 2
Indel Region Module

The main objective of this chapter is to describe the algorithm and output of the **indel region module**. SeqFIRE was designed to use different techniques to detect indels and conserved regions. The algorithm for indel identification and extraction is shown in Figure 2.1. The algorithm comprises several steps, which are explained below.

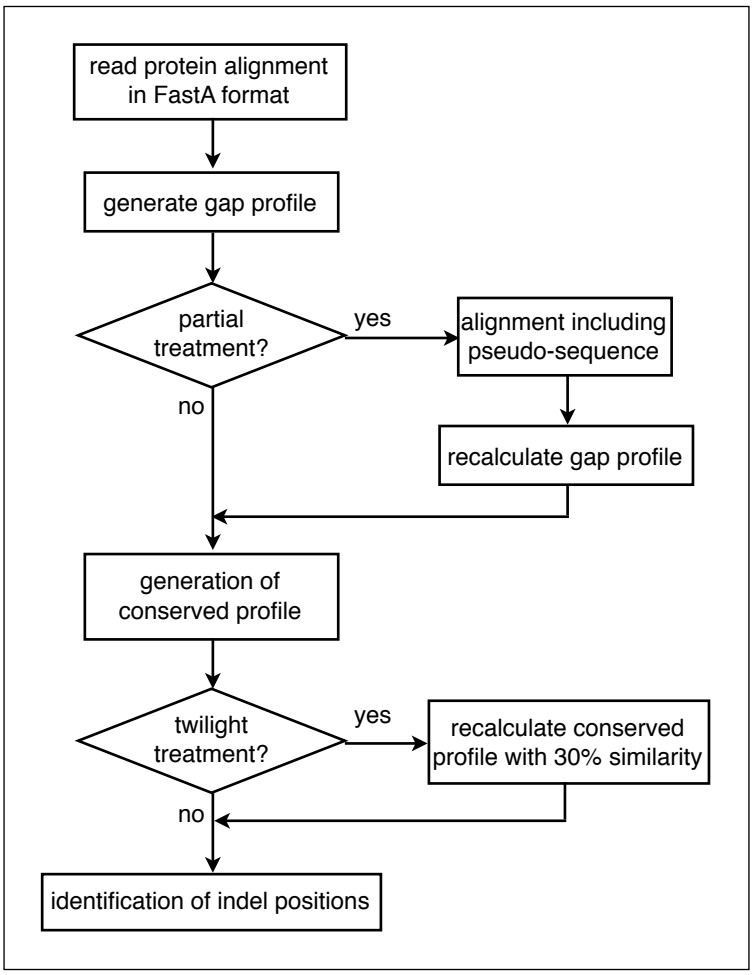


Figure 2.1 Workflow of the SeqFIRE indel region module (see text for details).

Structure of an Alignment

Orthologous proteins in different species often vary in length due to lineage-specific extensions and truncations of the protein during evolution. Such variations in length lead to indel-rich, and often uncertain alignment regions at the N- and C-termini. These regions are referred to here as **ragged regions** (due to their similarly to the ragged thread of an old cloth). The region in between the ragged regions is referred to as the **body of the alignment**. Inside the body of alignment, there may be multiple **non-indel regions** (also called conserved blocks) that are separated by **indel regions** (see

Figure 2.2). By definition all indel regions are flanked by one or more conserved alignment columns. The length of the conserved residues flanked indels determines the confidence of indel identification. The longer these regions are, the better.

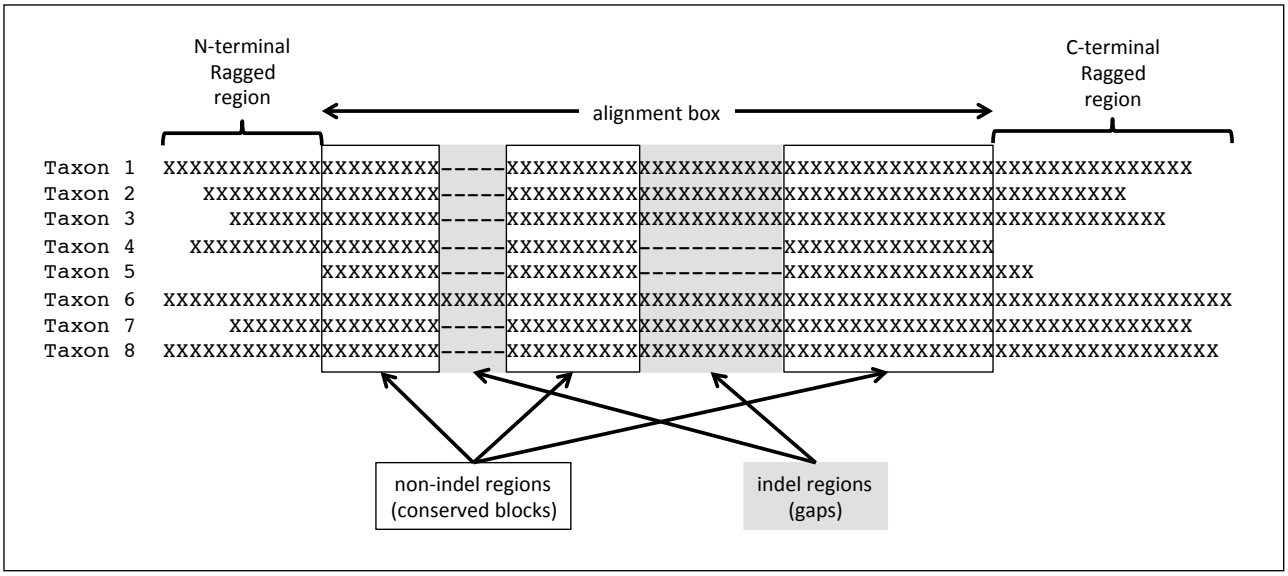


Figure 2.2 Designated regions of an alignment.
The braces at the left and right indicate the N- and C-terminal ragged regions, respectively. The white boxes indicate non-indel regions (conserved blocks), which are separated by indel regions (gray boxes).

Indel Region Module Algorithm

Generation of the Gap Profile

The SeqFIRE algorithm for identification and extraction of indel regions comprises several steps (see Figure 2.1). First, the program scans the alignment and generates a **gap profile**. This is a set of strings (one string per alignment column) containing binary values; '–' for gap containing positions (positions that have gap in any sequence at that position), and 'X' for gap free positions (positions that have no gap in any sequence at that position) (see Figure 2.3). The gap profile is the same length as the full alignment. So, positions in the gap profile correspond directly to positions in the alignment.

SeqFIRE uses the gap profile to define the ragged end regions of the alignment. SeqFIRE then eliminates the ragged regions and keeps the body of the alignment for the next step. Thus indels, as SeqFIRE defines them, are only found in the body of an alignment.

```
Taxon 1   ---------MVMKLYSKLQH----------------------DTSYKVVQLDDTILAAVKN---GEPLQFKSMDETQSEVVLCSSNA
Taxon 2   --------------------------MSINLHSAPEYDP-----SYKLIQLTPELLDIIQDPVQNHQLRFKSLDKDKSEVVLCSHDK
Taxon 3   -------MEKSSRIKGAESVLNLEPNSSIAIGYHALFGS---HDDLMLLEIDEKLFPDILH----ERVALRGQLDEDS--VLCTQSK
Taxon 4   -------MEEIGRIEGAKAVINLKPGSSVPISYHPCFGP---HEDLLLLEADDKLVSDIFH----ERVTLRGLPDEDA--VLCTKSK
Taxon 5   ----------MEEIGGAEAVINLKSGYSLPISYHPCFGP---HEDLLLLEADDKLVSDIFH----QRVTLRGLPDEDA--VLCTKSK
Taxon 6   --------RTVEDVDRILGFAKLTSTDLQPSVQCINFKSPIDNEAFKLMEMNEDMLRELED---GKKLVIRGDRADTA--VLCTKNK
Taxon 7   -MCDTDILSDIKDVRARLELAKLDIRDLKQPTQVLTFDEDANDQDVTLLELDKSVLQVIQN---GGSLVIRGNEDDTA--VLCTDDS
Taxon 8   MSVRVIPHRSQEEIFELLNFAKIDKNYMKNYVQSFYFGENLHHEDVYLFEIDKSLLEDLKS---SRSFVIRGGANDTA--VLCTESK


gap pro   ---------------------------------------------XXXXXXXXXXXXXXXXX----XXXXXXXXXXXXX--XXXXXXX
```

Figure 2.3 The gap profile.
Dash ('–') marks positions with a gap in at least one sequence. An 'x' marks position that has no gaps. The term 'gap pro' stands for gap profile.

Partial Treatment

However, ragged ends can also be the result of incomplete sequences. This is especially a problem with express sequence tag (EST) data or poorly annotated genomic data (incorrect start and/or stop codon identification). Incomplete sequences will also cause ragged end regions in an alignment, especially as EST sequences can be quite short. If you use the default settings with SeqFIRE, the program will truncate all ragged end regions, which could cause you to lose useful information, including interesting indels (see Figure 2.4).

In order to avoid losing potential information in the end regions of an alignment, SeqFIRE provides an option called **partial treatment**. Under this option, the program creates a new gap profile for ragged end regions by excluding the incomplete sequence(s) and using the remaining sequences to infer amino acid presence (but not identity) at unknown positions.

For example, in Figure 2.4, Taxon 5 contains an incomplete sequence. If the user chooses the partial treatment option for extraction of indels, SeqFIRE will exclude Taxon 5 (Figure 2.4A) and use the remaining sequences to generate a new gap profile for the ragged regions only. The program will then calculate a proportion of gap and non-gap characters at each position. If the position is >40% filled the position will be designated as non-gap at this position; if the percentage of non-gaps is lower than the 40% cut off, the program will treat that position will be merged as a gap and mark it as '–'.

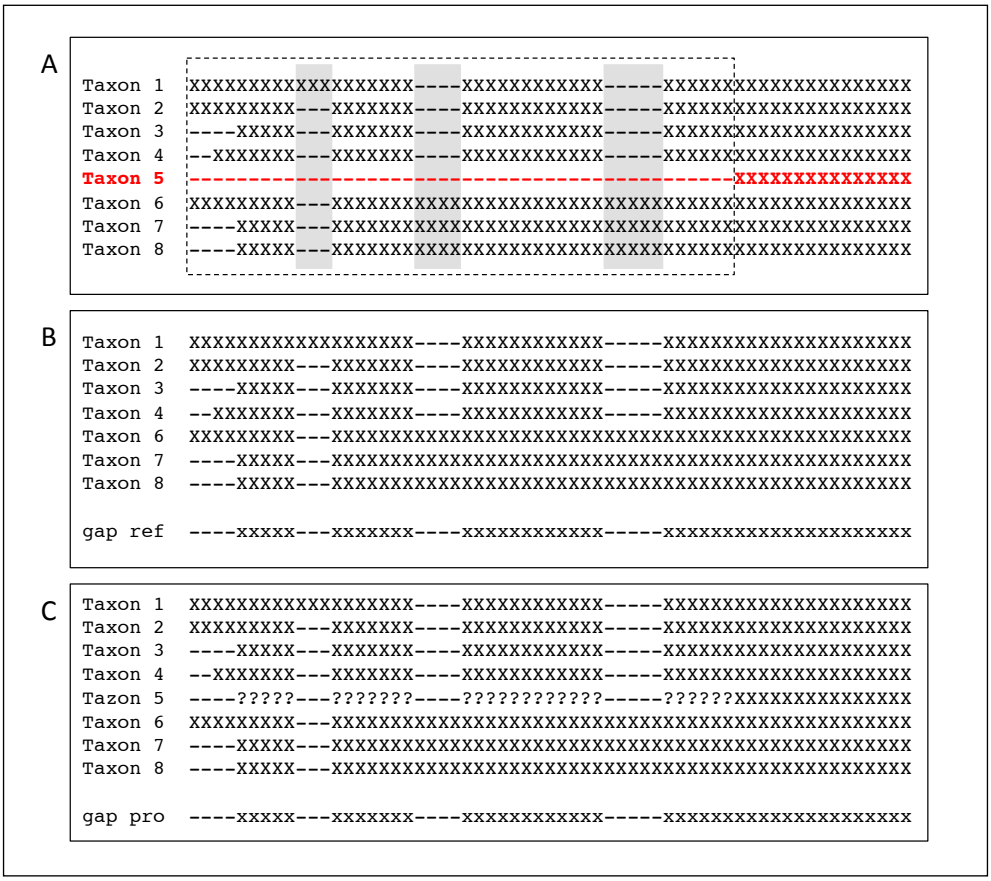


Figure 2.4 Partial treatment option.
The dash box shows an N-terminal ragged region of a protein alignment cause by the present of a partial sequence (Taxon 5). Within this ragged region, the gray boxes show indel regions that might be lost if the ragged region is truncated (A). The partial treatment option of SeqFIRE prevents the loss of this information by eliminating the incomplete sequence and then recalculating the gap profile (B). The program then uses the new gap profile as a guide to generate a pseudo-sequence (with "?") in the ragged region of the incomplete sequence (C).

Using this profile, SeqFIRE will then generate a **pseudo-alignment** for the partial sequence for the ragged region based on the new gap profile by adding the symbol "?" for every position marked as "x" in the new gap profile (Figure 2.4C).

If the program does not find any ragged regions, no modification will be made to the alignment even if the partial sequence treatment is chosen.

Generation of the Conservation Profile

The second guideline generate by SeqFIRE called **the conservation profile**. This is a string showing the conservation level of each alignment column based on a similarity score of amino acids in the column.

The similarity score can be based on sample amino acid identity using the default "NONE" setting. Alternatively SeqFIRE provides five sets of amino acid substitution groups based on PAM60, PAM250, BLOSUM40, BLOSUM62, and BLOSUM80 (Table 2.1). Amino acids in the same substitution group are recorded as identical when calculating the similarity score for an alignment column. The user can select a substitution model from the drop down menu under the **amino acid substitute groups** option (see Figure 1.2).

Table 2.1 Protein substitution models that are available in SeqFIRE.
Five different empirical matrices are available with amino acids arranged in groups of frequently substituted residues (Dayhoff *et al.*, 1978; Henikoff & Henikoff, 1992).

PAM60	PAM250	BLOSUM40	BLOSUM62	BLOSUM80
S, A, T	M, I, V, L	S, T	S, T	Q, R, K
H, N	D, N, H, Q, E	S, A	S, A	Q, E, K
S, N	F, I, L	S, Q, N	S, N	E, D
N, D	S, P, A	H, Y, M	H, Y	D, N
D, E	S, A, G	D, E	D, E	Q, H
H, Q	Q, K, N	N, D	N, D	Y, H
Q, E	R, H, Q	H, N	H, N	Y, W
Y, F	S, T	W, Y, F	W, Y, F	Y, F
R, K	R, K, Q	E, Q, K	E, Q, K	S, T
M, I	S, N	K, R	K, R, Q	S, A
L, M	F, W	R, Q	L, M, I, V	M, V, L, I
I, V	R, W	L, M, I, V		
	G, D	E, K		
		A, G		
		L, F, I		
		V, T		

SeqFIRE uses the chosen substitution model to calculate the % conservation for each alignment column. If the score is higher than the **conservation threshold** (which is defined by the user), SeqFIRE denotes the column as a "c" in the profile. Otherwise the program will mark the column as a "v" (see Figure 2.5). The user can adjust the conservation threshold by changing the amino acid conservation threshold (see Figure 1.2). The default value is 75% similarity. This value can be

increased if proteins in the alignment are highly conserved or decreased if proteins in the alignment are highly diverse.

Twilight Treatment

Protein structure has been shown to be more conservative than protein sequence (Illergård *et al.*, 2009). It has been found that proteins can still have the same structure even with sequence similarity as low as 25-35% (Rost, 1999). Therefore, not all homologous protein structures show high sequence similarity, and sometimes diverse protein sequences can share a conserved tertiary structure and function. This phenomenon is called the **protein twilight zone**.

To address this situation, SeqFIRE has an option called **twilight treatment** for low similarity homologous proteins. If this parameter is chosen, conservation threshold automatically be set to 30% similarity even if another value is defined by user in the conservation threshold setting (Figure 1.2). Then, SeqFIRE will replace the previous conservation profile with the new one (see Figure 2.5). If the twilight treatment is not chosen, this process will be skipped and no modification will be made to the conservation profile.

Generating the Indel Profile

In the last step in SeqFIRE's indel region module algorithm, an **indel profile** is generated using the conservation profile as a guideline. The indel profile is a string showing where indel regions occur in the alignment.

In order to identify the indel regions in an alignment, we have to make some basic assumptions about the indel region. An indel is the result of an insertion or deletion event in DNA. Because of this, each indel must be represented as a gap in the alignment. However, this can become problematic, as there may have been multiple genetic events that occurred in the same region, making the indel region appear increasingly complex. SeqFIRE uses these assumptions to setup four criteria for indel region identification.

(1) The indel region must contain a gap.
(2) The indel region must be flanked with conserved blocks consisting of at least three residues (default) on each side (conservation determined by conservation threshold and amino acid substitution group).
(3) All low similarity positions adjacent to the gap are identified as part of the indel region (Figure 2.5A).
(4) Any two indel regions must be separated by at least three conserved positions (default). Otherwise, both regions will be merged into a single indel region (see Figure 2.5B). The user can adjust the number of conserved sites between two indel regions by changing **inter-indel space** option on the input page (see Figure 1.2). *It is not recommended to set the inter-indel space to a number lower than three positions because any number below three greatly reduces the confidence of indel identification.*

SeqFIRE identifies indels by scanning the gap and conservation profiles from left to right and using the default or user defined criteria for flanking residues inter-indel space. When an indel region is encountered, the program will mark it with an "`I`" in the indel profile. Non-indel regions are denoted with a dot ("`.`") in the profile (Figure 2.5). If there are non-conserved positions adjacent to a gap region, those non-conserved positions will be included as part of the indel (Figure 2.5A). If the conserved

positions between two indels are less than the defined inter-indel space (default=3), the region will be marked as a single indel (Figure 2.5B).

```
                          A                              B

Taxon 1         XXXXXXXX-----XXXXXXXX        XXXXX----XX-----XXXXX
Taxon 2         XXXXXXXX-----XXXXXXXX        XXXXX----XX-----XXXXX
Taxon 3         XXXXXXXXXXXXXXXXXXXX        XXXXX----XX-----XXXXX
Taxon 4         XXXXXXXXXXXXXXXXXXXX        XXXXXXXXXXXXXXXXXXXX
Taxon 5         XXXXXXXXXXXXXXXXXXXX        XXXXXXXXXXXXXXXXXXXX

cons. profile   ccccccvv-----cccccccc        ccccc----cc-----ccccc
indel profile   ......IIIIIII........        .....IIIIIIIIIII.....
```

Figure 2.5 Brief criteria of indel recognition.
Non-conserved columns flanking gap columns are included in indel regions (A). Nearby indel regions are merged if separated by less than inter-indel space, which equals 3 by default (B).

Output Page

After SeqFIRE detects indels using the process described above, it will show the results on the output page (see Figure 2.6). The top section of the page gives links for jumping to the individual section of output:

(1) the alignment with indel profile in Jalview
(2) the alignment with indel annotation
(3) the indel list
(4) the indel matrix
(5) the masked indel alignment (see details below)

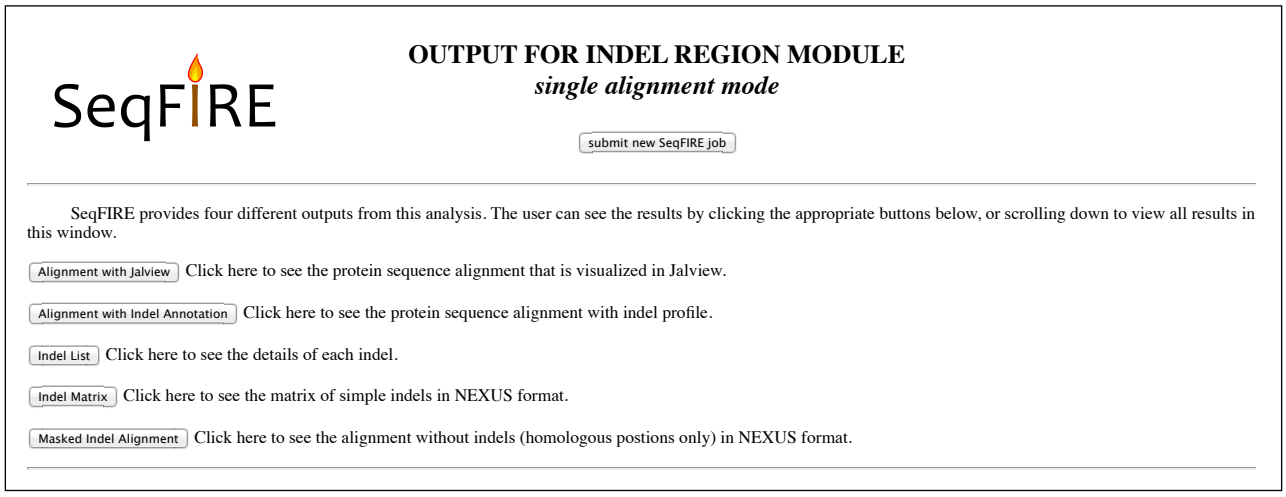


Figure 2.6 The top portion of the SeqFIRE output page.
At the top of page, SeqFIRE provides links for viewing all results including the protein alignment with the indel, conserved references in Jalview, the indel list, masked alignment, and NEXUS format of the indel matrix (if the program detects simple indels in and the alignment).

In addition, right-clicking on these links allows the users to directly download results (3) - (5). The middle section is the output graphically visualized using the Jalview, a multiple alignment editor.

The indel profile is shown under the alignment in Jalview, and the user can use any Jalview functions directly in SeqFIRE. The final section of the output is the full details of the remaining four outputs.

Alignment with Indel Profile in Jalview Visualization

SeqFIRE implements Jalview for visualization of the alignment together with the indel annotation (figure 2.7).

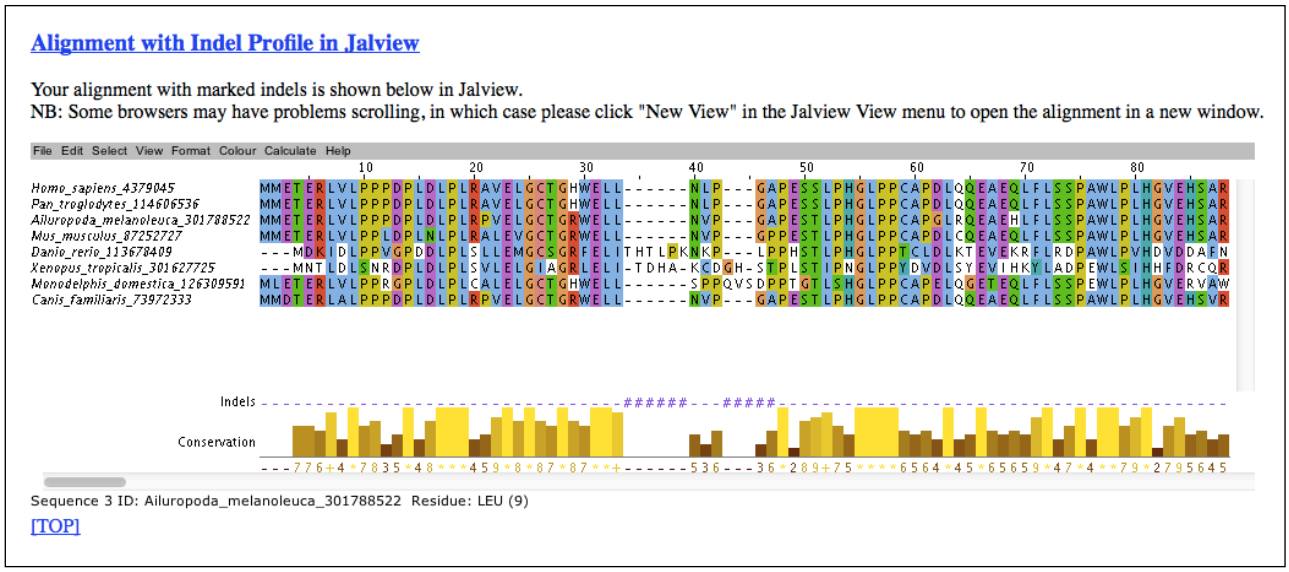


Figure 2.7 The alignment in Jalview with indel profile.
The indel profile is shown immediately below the alignment with indel regions designed by hashes "#".

Alignment with Indel Annotation

This output shows the full alignment with the indel regions indicated below in a simple format. A header provided listing all SeqFIRE parameter values plus the number of indels found in the alignment (Figure 2.8).

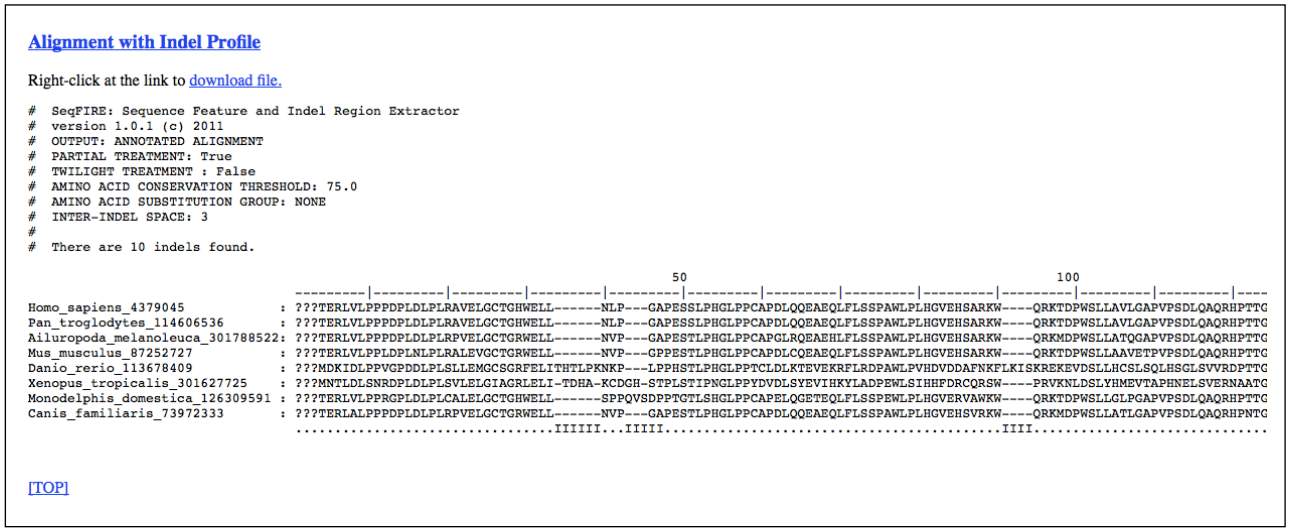


Figure 2.8 An example alignment with indel profile from the output page.

Indel List

The indel list output is a sequential display of all extracted indels. Each individual indel alignments is shown along with the flanking positions in the alignment. Each indel alignment is separated by double slashes ("//"). Each indel includes a header listing the indel number, location in the alignment, size (number of columns), and indel type (simple or complex). The total number of indel regions is reported in the header of the page below a list of the parameters used in defending the indels appears at the top of the list (Figure 2.9).

Indel List

Right-click at the link to download file.

```
#  SeqFIRE: Sequence Feature and Indel Region Extractor
#  version 1.0.1 (c) 2011
#  OUTPUT: INDEL LIST
#  PARTIAL TREATMENT: True
#  TWILIGHT TREATMENT : False
#  AMINO ACID CONSERVATION THRESHOLD: 75.0
#  AMINO ACID SUBSTITUTION GROUP: NONE
#  INTER-INDEL SPACE: 3
#
#  There are 10 indels found.
#  ** indicates masked sequence

//
Indel number: 1
Indel location in alignment: 34-39
Size of indel: 6 alignment columns
Type: complex indel

Homo_sapiens_4379045            : ELL  ------  NLP
Pan_troglodytes_114606536       : ELL  ------  NLP
Ailuropoda_melanoleuca_301788522: ELL  ------  NVP
Mus_musculus_87252727           : ELL  ------  NVP
Danio_rerio_113678409           : ELI  THTLPK  NKP
Xenopus_tropicalis_301627725    : ELI  -TDHA-  KCD
Monodelphis_domestica_126309591 : ELL  ------  SPP
Canis_familiaris_73972333       : ELL  ------  NVP
//
Indel number: 2
Indel location in alignment: 43-47
Size of indel: 5 alignment columns
Type: complex indel

Homo_sapiens_4379045            : NLP  ---GA  PES
Pan_troglodytes_114606536       : NLP  ---GA  PES
Ailuropoda_melanoleuca_301788522: NVP  ---GA  PES
Mus_musculus_87252727           : NVP  ---GP  PES
Danio_rerio_113678409           : NKP  ---LP  PHS
Xenopus_tropicalis_301627725    : KCD  GH-ST  PLS
Monodelphis_domestica_126309591 : SPP  QVSDP  PTG
Canis_familiaris_73972333       : NVP  ---GA  PES
//
Indel number: 3
Indel location in alignment: 91-94
```

Figure 2.9 Indel list output.
The list begins with a header listing all parameters and a total number of indels. This is followed by a sequential display of all indels.

Indel Matrix

Some indel regions are very clear and easy to recognize, because they contain only two states (present and absent). We call these "**simple indels**" because they appear to be the result of a single indel event. Some users may wish to use simple indels as binary characters (0/1 or preset/absent) in phylogenetic analyses. For this purpose, SeqFIRE parses the simple indels into a binary matrix in NEXUS format (see Figure 2.10).

The matrix contains a list of all sequence scored for the presence/absence of all simple indel regions identified in the alignment. If an individual sequence contains residues at an index location, the indel will be marked as "1" (present) in the matrix for that particular sequence. If there are no residues in that location, it will be marked as "0" (absent).

The file also included a list of all indels with their size and location in the alignment at the bottom of the file. The list is printed in a "notes block" in blackest so it will not interfere with execution of the file. Therefore, the file can be executed on its own or pasted into an already existing NEXUS file.

Indel Matrix in NEXUS Format

Right-click at the link to download file.

```
#NEXUS
BEGIN DATA;
    DIMENSION NTAX=8 NCHAR=5;
    FORMAT DATATYPE=SYMBOL "0 1";
    OPTIONS GAPMODE=MISSING;

MATRIX
[                                      ]
[                                 -----]
Homo_sapiens_4379045            : 01111
Pan_troglodytes_114606536       : 01111
Ailuropoda_melanoleuca_301788522: 01111
Mus_musculus_87252727           : 01111
Danio_rerio_113678409           : 11000
Xenopus_tropicalis_301627725    : 00111
Monodelphis_domestica_126309591 : 01111
Canis_familiaris_73972333       : 01111
;
END;

BEGIN NOTES;
[ Indel Number     Alignment Position      Indel Length ]
[ ------------     ------------------      ------------ ]
[     3                91-94                   4        ]
[     6                556                     1        ]
[     7                566-578                 13       ]
[     8                787-788                 2        ]
[     10               1042-1046               5        ]

END;
```

[TOP]

Figure 2.10 Indel matrix output.
A list of all sequences given in NEXUS format. This is followed by a complete annotated list of indels given in a "notes block" so that it will not interfere with execution of the file.

Masked Indel Alignment (Alignment without Indels)

SeqFIRE also provides the alignment without indel regions in NEXUS format. All indel regions identified are removed so that the remaining alignment consists only regions with confident alignment. The result is in NEXUS format and is suitable to uses input for phylogenetic analysis (Figure 2.11).

Masked Indel Alignment in NEXUS Format

NOTE: This output was generated by deselecting the partial treatment option automatically.
Right-click at the link to download file.

```
#NEXUS
BEGIN DATA;
     DIMENSIONS NTAX=8 NCHAR=1202;
     FORMAT MISSING=? DATATYPE=PROTEIN GAP=-;
     OPTIONS GAPMODE=MISSING;

MATRIX
[                                                                                  50                                                  100
[                                  ---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---
Homo_sapiens_4379045               TERLVLPPPDPLDLPLRAVELGCTGHWELLNLPPESSLPHGLPPCAPDLQQEAEQLFLSSPAWLPLHGVEHSARKWQRKTDPWSLLAVLGAPVPSDLQAQRHPTTGQILGYKEVLLENTNLS
Pan_troglodytes_114606536          TERLVLPPPDPLDLPLRAVELGCTGHWELLNLPPESSLPHGLPPCAPDLQQEAEQLFLSSPAWLPLHGVEHSARKWQRKTDPWSLLAVLGAPVPSDLQAQRHPTTGQILGYKEVLLENTNLS
Ailuropoda_melanoleuca_301788522   TERLVLPPPDPLDLPLRPVELGCTGRWELLNVPPESTLPHGLPPCAPGLRQEAEHLFLSSPAWLPLHGVEHSARKWQRKMDPWSLLATQGAPVPSDLQAQRHPTTGQILGYKEVLLENTNLS
Mus_musculus_87252727              TERLVLPPLDPLNLPLRALEVGCTGRWELLNVPPESTLPHGLPPCAPDLCQEAEQLFLSSPAWLPLHGVEHSARKWQRKTDPWSLLAAVETPVPSDLQAQRHPTTGHILGYKEVLLENTNLS
Danio_rerio_113678409              MDKIDLPPVGPDDLPLSLLEMGCSGRFELINKPPHSTLPHGLPPTCLDLKTEVEKRFLRDPAWLPVHDVDDAFNKFKREKEVDSLLHCSLSQLHSGLSVVRDPTTGMLLDFTEVLLEDTGLS
Xenopus_tropicalis_301627725       MNTLDLSNRDPLDLPLSVLELGIAGRLELIKCDPLSTIPNGLPPYDVDLSYEVIHKYLADPEWLSIHHFDRCQRSWPRVKNLDSLYHMEVTAPHNELSVERNAATGELHGINETSVDHCSLS
Monodelphis_domestica_126309591    TERLVLPPRGPLDLPLCALELGCTGHWELLSPPPTGTLSHGLPPCAPELQGETEQLFLSSPEWLPLHGVERVAWKWQRKTDPWSLLGLPGAPVPSDLQAQRHPTTGRVLGYREVLLENTNLS
Canis_familiaris_73972333          TERLALPPPDPLDLPLRPVELGCTGRWELLNVPPESTLPHGLPPCAPDLQQEAEQLFLSSPAWLPLHGVEHSVRKWQRKMDPWSLLATLGAPVPSDLQAQRHPNTGQILGYKEVLLENTNLS
;
END;
```

[TOP]

Figure 2.11 The masked indel alignment.
The alignment with all indel regions removed is output in NEXUS format.

Chapter 3

Conserved Block Module

The **conserved block module** can be used for identification and extraction of conserved blocks in an alignment for subsequent use in other applications, *e.g.* phylogenetic analysis. SeqFIRE uses a combination of similarity scoring and information entropy scoring techniques to determine conserved regions. A flow chart of the conserved block module is shown in Figure 3.1 (see the text below for detalis). The user can also extract the conserved blocks and the indel matrix simultaneously within this module.

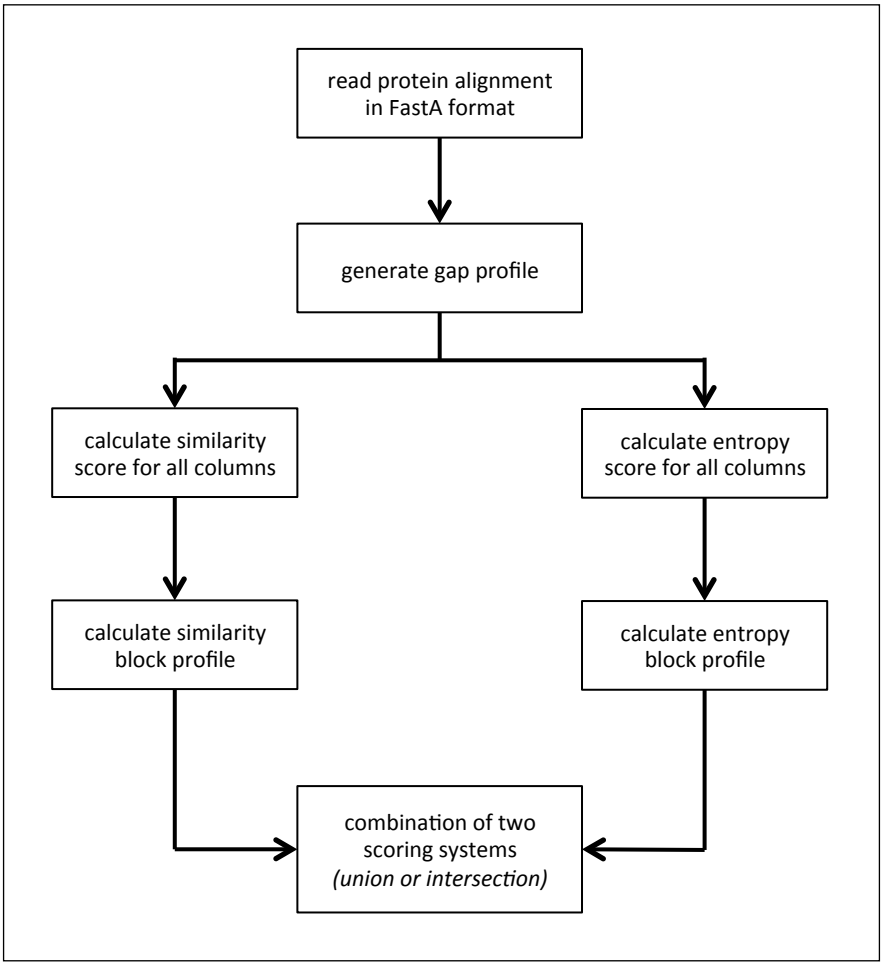


Figure 3.1 Workflow of conserved block module (see text for details).

Similarity and Information Entropy

SeqFIRE uses information from two different kinds of scanning methods to define conserved sequence blocks in an alignment. The first and simplest method scores the degree of divergence among sequences using an identity score. The amino acid alignment identity score is calculated by identifying the most abundant amino acid (letter) in each position (column) of the alignment. The percentage frequency of the most frequent amino acid is determined, and this is the **similarity score** of that position.

However, some amino acid substitutions are more frequent than others, *e.g.* because they do not change the protein structure and/or because they have similar physicochemical properties. To reflect this, the identity scoring method has been modified so that amino acids that share the same physicochemical properties are counted as the same state.

The second scoring method employed by SeqFIRE is the **information entropy**, also called **Shannon's entropy**. This is another index to measure the degree of diversification in the data. The term "**entropy**" in thermodynamics means disorder, but in the alignment, entropy also means **variation** (disorder) of amino acids in a particular position of the alignment. A way to measure this variation in the data is the information entropy (H).

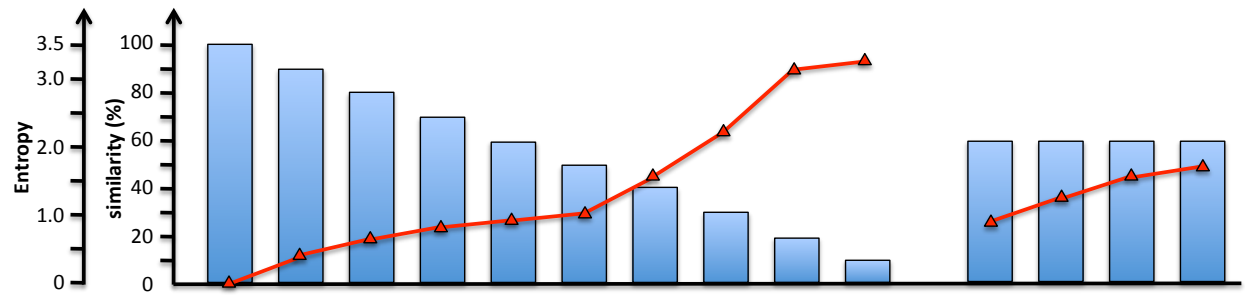
SeqFIRE calculates the entropy for each alignment column using the equation of Weaver and Shannon (1963) using the following equation:

$$H=-\sum_{i=1}^{20} p_i log\ p_i \tag{1}$$

where p_i is the proportion of amino acid i in a particular column of the alignment. For example, consider column two of the alignment in Figure 3.2: there are nine aspartates (D_s) in the column for 10 sequences. Given this, the proportion for aspartate p_D is 9/10 or 0.9. There is also one glutamate (E) in the same column so the probability of glutamate p_E is 1/10 or 0.1. From equation (1), we get

$$H=-[(0.9 log\ 0.9)+(0.1 log\ 0.1)] \tag{2}$$

So, the information entropy (H) in this column is 0.1412. High entropy implies high diversification in the alignment position, whereas low entropy indicates a high level of conservation in the alignment position (Figure 3.2).



Taxon_1	L	D	Y	L	N	L	C	A	M	Q	K	V	W	C
Taxon_2	L	E	Y	L	N	V	C	A	M	S	K	D	Y	M
Taxon_3	L	D	F	L	N	V	C	A	V	T	K	D	N	Q
Taxon_4	L	D	F	I	N	V	C	S	H	W	K	D	N	F
Taxon_5	L	D	F	I	S	V	P	S	K	F	V	K	H	D
Taxon_6	L	D	F	I	S	V	P	I	N	V	V	K	H	D
Taxon_7	L	D	F	I	S	L	Y	R	A	H	V	K	H	D
Taxon_8	L	D	F	I	S	L	Y	R	P	E	V	K	H	D
Taxon_9	L	D	F	I	S	L	Y	D	K	N	V	K	H	D
Taxon_10	L	D	F	I	S	L	P	D	Q	Y	V	K	H	D

Figure 3.2 Correlation between similarity score (bars) and information entropy (lines).

Similarity values show inverted values relative to information entropy (Figure 3.2). This is a highly conserved position will show a high similarity score but low entropy, and a highly diverse position will have a lower similarity score but high entropy.

Conserved Block Module Algorithm

Generation of the Gap Profile

In the conserved region module, SeqFIRE starts by creating a **gap profile**. This is different from the gap profile in the indel module. Here, the proportion of gap and non-gap characters is calculated for each position in the alignment. If the proportion of gap characters is larger than or equal to 40% (default), SeqFIRE will treat this column as a gap and mark it with a "–" in the gap profile. If the proportion is less than 40% (default), the program will classify this column as a non-gap, and mark an 'x' in the gap profile. The user can adjust the threshold for gap/non-gap acceptance by changing the number in the option called "**percentage of gaps accepted**" on the input page.

Conserved Block Identification Using Similarity Scoring

Based on the gap profile, all positions marked with an "x" will then be scored for similarity. This can be done with all substitution weight equally or the user can choose to apply a substitution weight matrix to the calculation. SeqFIRE provides five substitution model options (see Table 2.1) as described in Chapter 2 (*Generation of the Conservation Profile*). If the program finds a similarity score equal to or higher than 75% similarity (default), the symbol "c" will be marked in the **similarity profile**. Otherwise, SeqFIRE will mark a "v" (variable) in the similarity profile. SeqFIRE will skip this calculation for positions designated as gap, and mark a "–" in the similarity profile to indicate a gap character.

After the similarity profile is generated, the program will modify this profile in order to find conserved blocks using a similarity-based method as follows. Firstly, all non-conserved regions (regions marked as "v" in the similarity profile) that are less than three continuous characters long (default) will be merged to be included in the flanking conserved block (see Figure 3.3). The user can adjust this parameter by changing the number in the **maximum size of non-conserved block** option (see Figure 1.4). Regions consisting of three or more contiguous will not excluded from the conserved block.

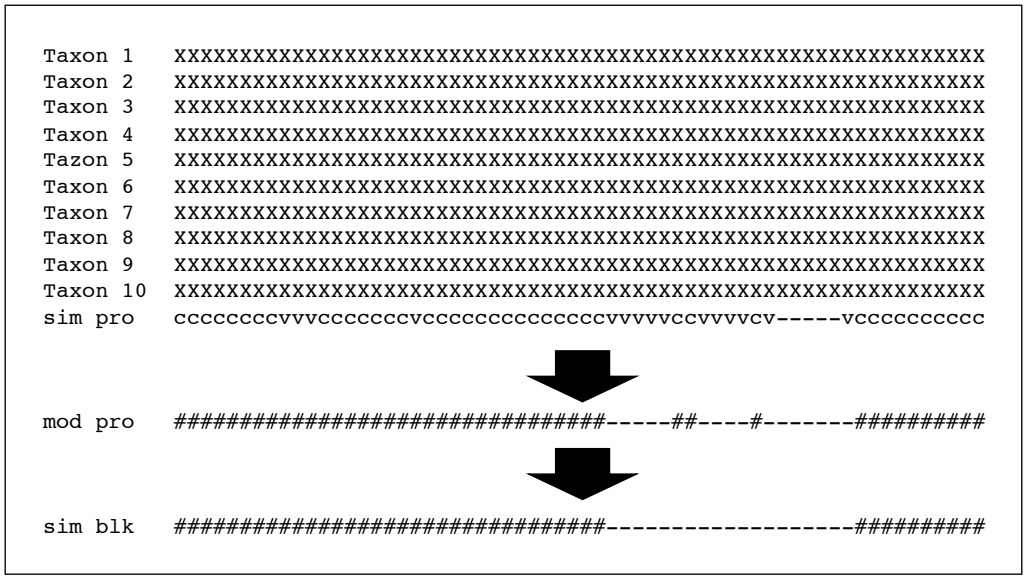


Figure 3.3 Steps in generation of the similarity block profile.
The method starts with modification of the similarity profile using two parameters: **maximum size of the non-conserved blocks** and **minimum size of conserved blocks** (see detail in the text below). The words "Sim pro" , "mod pro", and "sim blk" stand for similarity profile, modified similarity profile, and similarity block, respectively.

Every position that SeqFIRE assigns to a conserved block will be marked as "#" in the similarity block profile. Then, every conserved block of less than three characters (default) will be excluded. This default value of three can be adjusted by changing the number in the **minimum size of conserved block** option (see Figure 1.4). The final profile after all these modifications is called the **similarity-block profile**.

Conserved Block Identification Using Entropy Scoring

SeqFIRE also provides another technique for predicting conserved blocks. The algorithm starts by calculating the information entropy for all non-gap positions in the alignment using equation (1). Once SeqFIRE has found the entropy values for all positions, it will calculate an overall **cut off** for conserved region identification. This is set to the median site by site entropy score plus standard deviation (*SD*), according to the following equation:

$$cut\ off = Median\ of\ entropy \times 1.4826 \tag{3}$$

All positions with an equal or lower value than the cutoff threshold will be marked as "c", and all positions that have a score higher than the cut off will be marked as "v". Conserved blocks are then identified using the same algorithm as the conserved block identification using similarity scoring (See: previous section).

All regions that have strings of "v"s shorter than three characters in length (default **maximum size of non-conserved block**) are merged with their flanking conserved blocks (marked with a "#" in alignment profile). The program will assigns all conserved blocks shorter than three characters, as part of their adjacent gap regions. The final version of the conserved block profile is called the **entropy-block profile**.

Caution: *These two parameters (**maximum size of non-conserved block** and **minimum size of conserved block**) are shared with the previous method and cannot be independently adjusted for the different conserved block identification methods.*

Combining the Conserved Blocks from the Two Scoring Techniques

Final step of the conserved block identification is the combination of the two conserved block profiles: similarity and entropy-block profiles. SeqFIRE has two options for combination those profiles, based on the basic mathematics (union and intersection). The **union** method is a relax combination. The program will mark "#" in the final conserved block if that position in either similarity-block or entropy-block profiles has "#". Intersection method is more restricted. The program will mark "#" in the final conserved block if and only if the "#" is scored in the same position in both similarity-block and entropy-block profiles.

Note: We suggest user uses intersection in case of the identification of highly conserved blocks, and uses union in general case, particularly if there is a highly diverse sequence in the alignment.

Output Page

SeqFIRE provides six different outputs. The first two outputs are the whole alignment with conserved block annotation that is visualized separately in Jalview and text mode. The next two outputs are in FastA format; first is the sequence alignment with conserved block profile, and second

the alignment of conserved blocks only (un-conserved block regions deleted). The last two outputs are in NEXUS format. First the alignment with its conserved block profile in a hidden (brackets) comment line and second the indel masked alignment (alignment with indel regions removed).

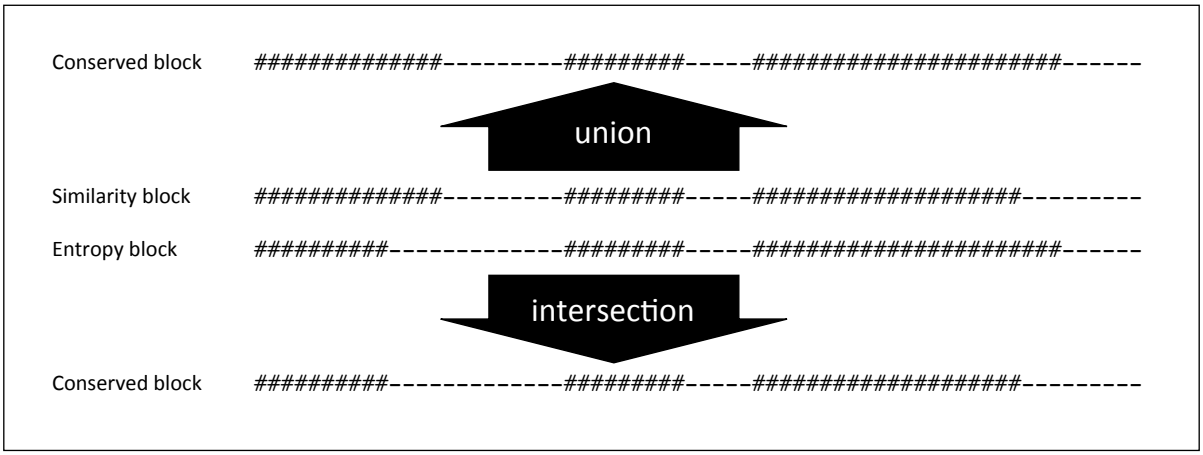


Figure 3.4 Two methods for combination of similarity- and entropy-block profile.
SeqFIRE scans both profiles in each position. If one or both profiles show a "#" in the profile, the program will mark "#" in the final conserved block.

Co-analysis of Conserved Blocks and Indel Regions

Some users may want to build a phylogeny using both the sequence data and the binomial character data. This is possible by co-analysis of the indel regions and the conserved block modules. In order to do this, first go to the conserved block page and select your parameters. Before clicking the FIRE! button, scroll down to the bottom of the page. Here, you can see the section "**Co-analysis with indel region module**". In this section, you can choose the option **Co-analysis with indel region module** (see Figure 3.5). The default of this option is "**Use conserved block module alone**", in which the indel region co-analysis option will automatically appear in the output section. Then user can then select the parameters for the indel region module, and click FIRE!. The program will execute both module simultaneously. If the program finds at least one simple indel, the resulting NEXUS file will include both conserved blocks and an index matrix.

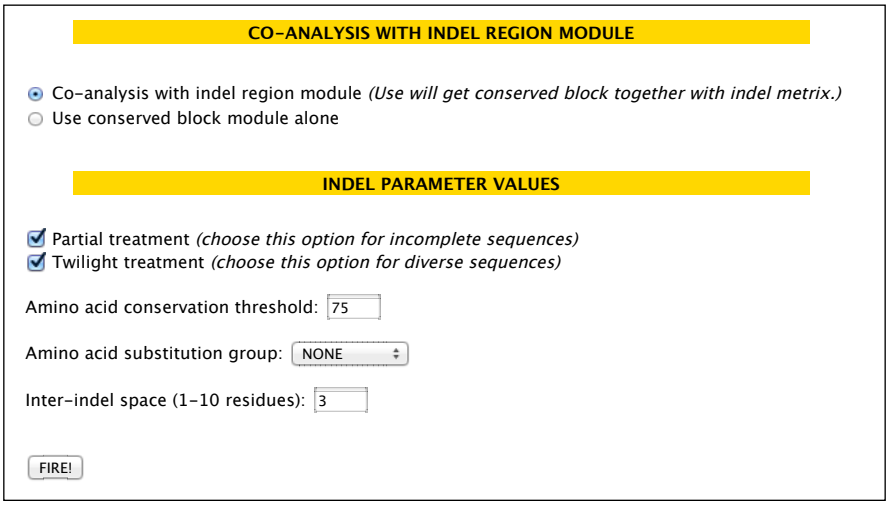


Figure 3.5 The co-analysis with indel region module section.
The co-analysis with indel region module section at the bottom of the conserved block module page allows simultaneous analysis of indels and conserved blocks.

Chapter 4

Working with Multiple Datasets

This chapter is for users who want to run SeqFIRE with a large amount of data (several sequence alignments). We provide a **batch mode analysis** option for both the indel region and conserved block modules. In order to use the batch mode, first prepare the input files in a SeqFIRE compatible format. For this, we provide a small Python script called **SeqFIREprep**, which can be downloaded from the SeqFIRE web server under the **Download** tab. SeqFIREprep will merge multiple alignment files into a single large input file. The script can also be used for subsequent separation of the results into separate output files for each alignment.

Installation of SeqFIREprep

Users need the Python interpreter to run SeqFIREprep. You can download the Python interpreter from the official Python website (www.python.org/download/). SeqFIREprep works well on Python interpreter versions 2.6 and above.

Installation of SeqFIREprep is very easy. After installing the Python interpreter, copy SeqFIREprep into an accessible folder and then SeqFIREprep is ready to be implemented in command line format. For Windows users, you can launch the **Command Prompt** using the start button. For Mac users, open a **Terminal** from the Application folder. For Linux users, launch the **Terminal** at the menu bar.

Preparation of the Input Data

Once the terminal or command prompt is launched, move to the directory where SeqFIREprep is installed. Type the command:

```
>>> python seqfireprep.py
```

You will see the menu as shown in the Figure 4.1.

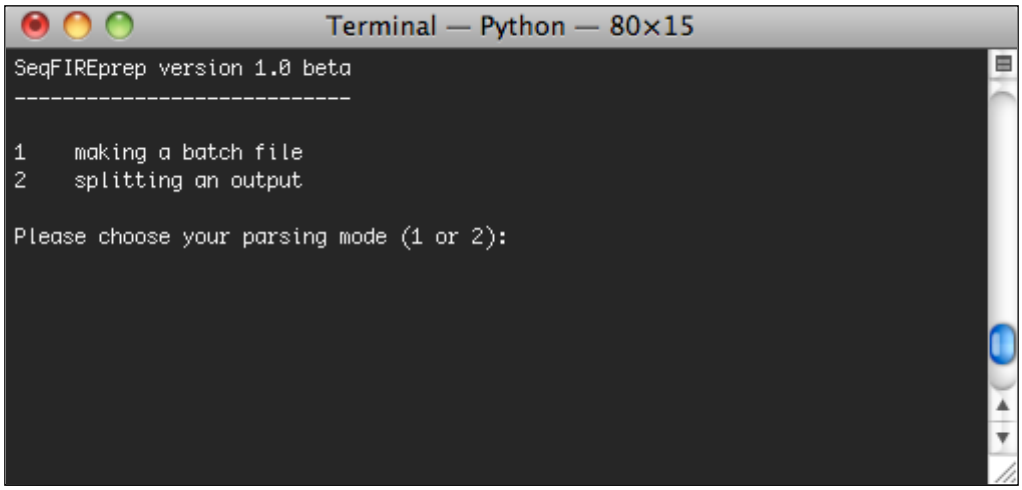


Figure 4.1 Main menu of SeqFIREprep.

To combine multiple input files, type 1 and hit <return>. The new menu will display the following (Figure 4.2):

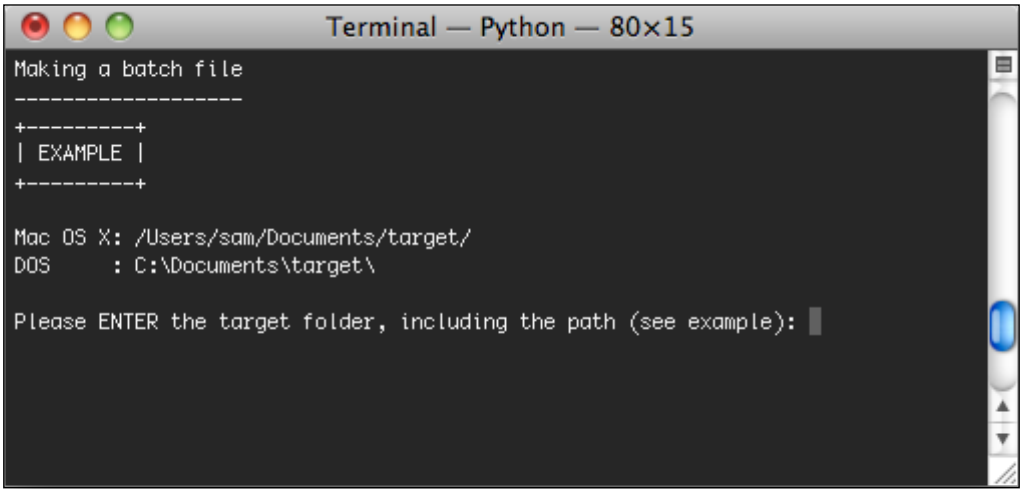


Figure 4.2 Assigning the destination folder.

Enter the destination folder name (or address). SeqFIREprep will automatically combine all files in that folder into a single input file, called "**batch.fa**". The SeqFIREprep format for the combined file begins with the filename on the first line, and is followed by all data within that file. The filename is flanked with "`==seq==`" and "`==fire==`" (see Figure 4.3).

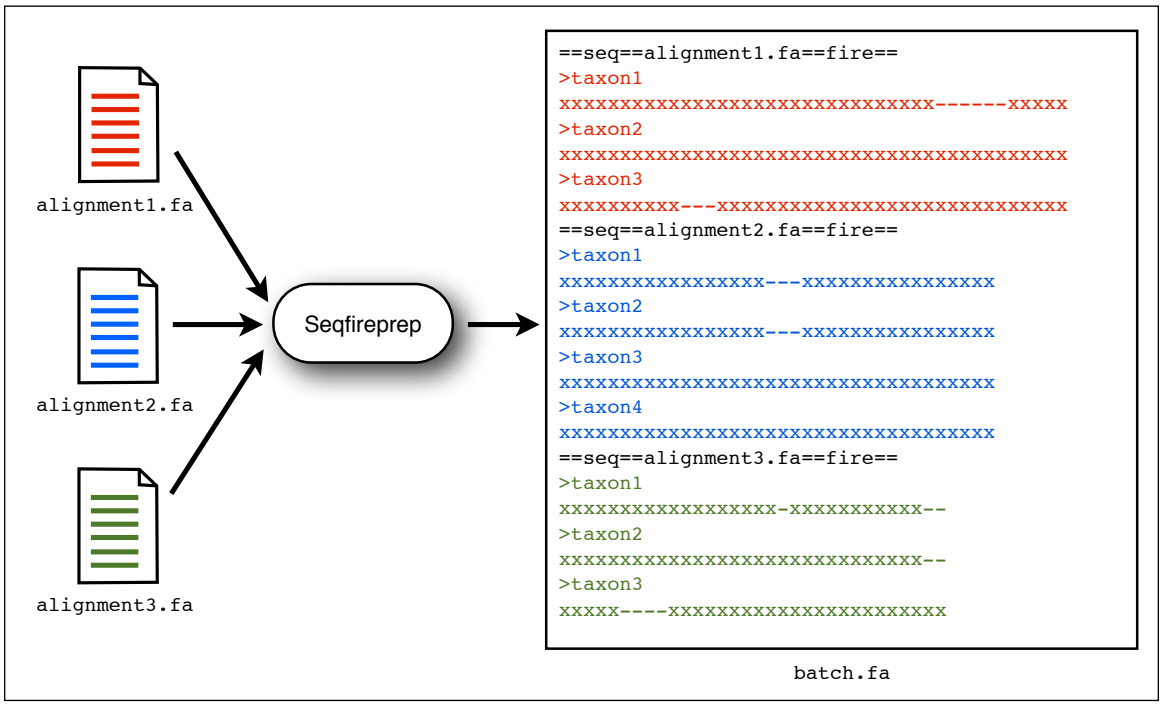


Figure 4.3 A flowchart showing how the file "batch.fa" is generated and the format of the resulting batch.fa file.

Using the Batch Mode

In order to use the batch mode for the indel region and conserved block modules, the user must select the batch mode button on the SeqFIRE top page (see Figure 1.1). This will take you to the batch mode input page (see Figure 4.4).

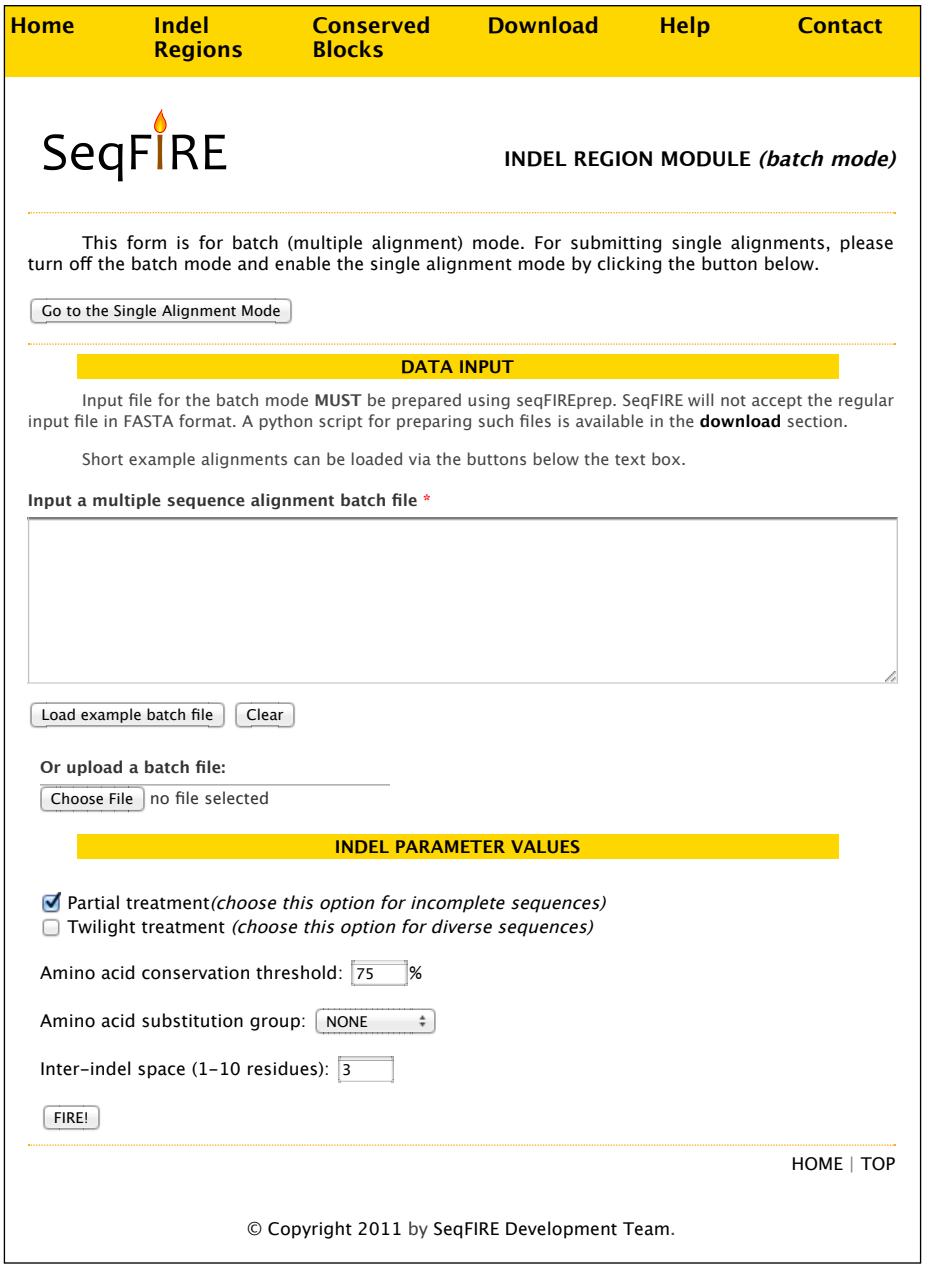


Figure 4.4 Indel region module batch mode input screen.

Users can either upload the batch file, or copy and paste it into the input box. However, SeqFIRE will only accept input prepared by SeqFIREprep or in the equivalent format (Figure 4.3). Then, the user can select parameters like a normal analysis. The selected parameters will apply for all input data.

Separation of the Outputs

After the analysis is completed the output for each input alignment can be separated, again using SeqFIREprep. Once the terminal or command prompt is launched, move to the directory where SeqFIREprep is installed. Then, launch the script as follows:

```
>>> python seqfireprep.py
```

The same SeqFIREprep menu will appear as in the Figure 4.2. This time select option 2 and enter. In the new screen that will appear, enter the name of the result file (Figure 4.5).

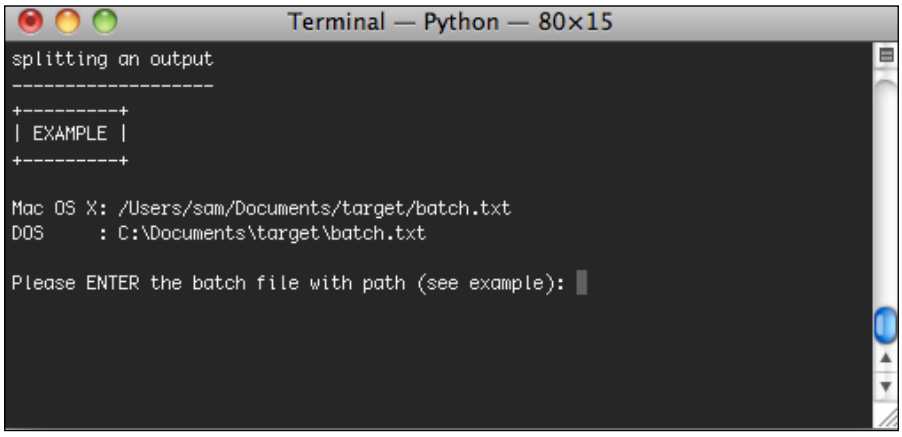


Figure 4.5 SeqFIREprep asks for the destination of the result file.

The output from SeqFIRE, SeqFIREprep will then automatically separate the result into separate files for each input alignment. The resulting files will be given the same names as the initial input file.

Chapter 5
Running SeqFIRE Locally

This chapter provides details of SeqFIRE's options for users who want to run SeqFIRE on a standalone computer or a local server. You can download the SeqFIRE standalone version directly from the download section of the web server (http://www.seqfire.org/seqfire_download.html).

Installation

Just as with SeqFIREprep, the standalone version of SeqFIRE needs the Python interpreter to run the script. To install the Python interpreter see **installation** in Chapter 4. Then copy standalone SeqFIRE into an accessible folder. SeqFIRE can then be implemented in command line format. Windows users can do this by launch the **Command Prompt** using the start button. For Mac users, open a **Terminal** (Application folder), and Linux users should launch a new **Terminal** from the menu bar.

General Options

Help Option (-h)

This command will show all parameters that are available for running standalone SeqFIRE.

Format

```
>>> python seqfire.py -h
```

Input Option (-i)

The user can specify the input file, including the directory of the input file using the `-i` option. If the input file is located in the same directory as the SeqFIRE program, the user can simply specify the filename. This is a compulsory option. If this option is skipped, SeqFIRE will instead look for an input file with the name "**infile.fa**" in the SeqFIRE directory.

Format

```
>>> python seqfire.py -i {filename}
```

Examples

```
>>> python seqfire.py -i /Users/Pravech/desktop/inputfile.fa
```

or

```
>>> python seqfire.py -i inputfile.fa
```

Analysis Mode Option (-a)

All analyses either the indel module or the conserved block module are run via the `-a` option. If the `-a` option is "1", SeqFIRE will process the input in the indel module. If "2" is selected, SeqFIRE will run the input file with the conserved block module. The default option is "1". To run SeqFIRE in both modules (indel regions & conserved blocks together), set -a to "2" (see: Co-analysis Option page 34)

Format

```
>>> python seqfire.py -i inputfile.fa -a {1 or 2}
```

Examples

```
    >>> python seqfire.py -i inputfile.fa -a 1      (indel module)
or  >>> python seqfire.py -i inputfile.fa -a 2      (conserved block module)
```

Output Option (-o)

SeqFIRE allows three different output options. To see the result on the screen only, type 1 after `-o`. To get the results in an output file only, type 2. To get the results both on screen and in an output file type 3. The default option is 2.

Format

```
>>> python seqfire.py -i inputfile.fa -o {1, 2, or 3}
```

Examples

```
    >>> python seqfire.py -i inputfile.fa -o 1      (output to screen)
or  >>> python seqfire.py -i inputfile.fa -o 2      (output to file)
or  >>> python seqfire.py -i inputfile.fa -o 3      (output to screen and file)
```

Indel Region Module Options

Similarity Threshold Option (-c)

This option sets the similarity threshold for scoring the conserved positions in the indel region module only. The similarity threshold is set at a given percent after the option `-c`. The default threshold is 75. The threshold input can either be in integer or real (decimal) numbers.

Format

```
>>> python seqfire.py -i inputfile.fa -c {similarity score}
```

Examples

```
>>> python seqfire.py -i inputfile.fa -a 1 -c 80.5
```

or **>>> python seqfire.py -i inputfile.fa -a 1 -c 70**

Substitution Group Option (-g)

Substitution matrix can be selected in order to modify the identity scores for conserved positions in the indel region module according to a set of evolutionary models. There are six different options: NONE, PAM60, PAM250, BLOSUM40, BLOSUM62 and BLOSUM80 (see chapter 2 for more detail). If NONE is selected, SeqFIRE will automatically calculate the identity score instead of the similarity score (identity × substitution model). The default option is NONE.

Format

>>> python seqfire.py -i inputfile.fa -g *{matrix}*

Examples

>>> python seqfire.py -i inputfile.fa -a 1 -g NONE
or **>>> python seqfire.py -i inputfile.fa -a 1 -g PAM250**

Inter-indel Space Option (-b)

This parameter determines the minimum space (number of conserved alignment columns) separating two indel regions. For example, if this parameter is set to three, all indel regions separated by less than three conserved columns (=1 or 2) will be merged into a single indel region. As a result, only indel regions will be separated by at least three alignment columns will be recognized as unique. The default option is three columns. The space between indel regions must be an integer value.

Format

>>> python seqfire.py -i inputfile.fa -b *{space between indels}*

Examples

>>> python seqfire.py -i inputfile.fa -a 1 -b 2
or **>>> python seqfire.py -i inputfile.fa -a 1 -b 10**

Partial Treatment Option (-p)

If the input data contains incomplete sequences, SeqFIRE will automatically truncate the overhanging sequences at both the N- and C- termini of the alignment. This action could result in the loss of informative indels. To avoid this, the partial sequence treatment option allows the program to retain overhang regions in the analysis by identifying partial sequences and filling them in with a consensus (see section…..). To invoke this setting, use the `-p` option with "True" (include overhang

regions in the analysis) or "False" (discard incomplete terminal regions). The default setting is True. The implications of this setting are further discussed in the **Partial Treatment** section in Chapter 2.

Format

```
>>> python seqfire.py -i inputfile.fa -p {True or False}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 1 -p False
```

Twilight Treatment Option (-t)

Twilight treatment deals with homologous proteins that may have a similar structure despite very low sequence similarity. Using the default similarity cut off (75%), conserved sites will be difficult to identify, and indel regions will tend to be merged. If your alignment has one or more very divergent sequence(s), we suggested trying the twilight treatment. This can be done using the `-t` option with "True". If your data set contains highly conserved sequences, this option should be turned off. To turn this option off, type "False" after `-t`. The default option is False. The implications of this setting are further discussed in the **Twilight Treatment** section in Chapter 2.

Format

```
>>> python seqfire.py -i inputfile.fa -t {True or False}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 1 -t False
```

Options for the Conserved Block Module

Percent Accept Gap Option (-j)

This option sets the similarity threshold for scoring conserved positions in the conserved profile used to define blocks. This similarity score (in percent) is set after the option `-j`. The default threshold is 75. The threshold value can either be in integers or real (decimal) numbers.

Format

```
>>> python seqfire.py -i inputfile.fa -j {percentage}
```

Examples

```
>>> python seqfire.py -i inputfile.fa -a 2 -j 40.0
```

or

```
>>> python seqfire.py -i inputfile.fa -a 2 -j 45
```

Similarity Threshold Option (-d)

In the similarity-block profile, the user has also set a threshold cut off for the identification of conserved position. This is similar to the `-c` option in the indel region module (page xx). The default for `-d` is 75%. Note that SeqFIRE allows the user to set different similarity thresholds in the indel module and the conserved block module (see: Co-analysis, page xxx, for more information). The threshold input can either be an integer or real (decimal) number.

Format

```
>>> python seqfire.py -i inputfile.fa -d {similarity score}
```

Examples

```
>>> python seqfire.py -i inputfile.fa -a 1 -d 75.5
```

or

```
>>> python seqfire.py -i inputfile.fa -a 1 -d 60
```

Substitution Group Option (-k)

This option determines the amino acid substitution group used in calculating the similarity score. There are six choice are NONE, PAM60, PAM250, BLOSUM40, BLOSUM62 and BLOSUM80. This option is for the conserved block module only. The default is NONE.

Format

```
>>> python seqfire.py -i inputfile.fa -k {matrix}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -k BLOSUM62
```

Minimum Space between Two Blocks Option (-s)

SeqFIRE uses this parameter in generating similarity and entropy profiles. SeqFIRE uses the similarity and entropy profiles using parameters defined by `-j`, `-d` and `-k` to define the limits of conserved blocks (see Chapter 3). The program then deletes all blocks that are equal in length or shorter than the `-s` value defined here (default = 3). This parameter must be adjusted together with the maximum size for non-conserved blocks (see the next section).

Format

```
>>> python seqfire.py -i inputfile.fa -s {number of residues}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -s 3
```

Maximum Size for Non-conserved Block Option (-f)

This option is rerated to the minimum size of conserved block option. Once SeqFIRE has eliminated the small conserved blocks (based on `-k` set above), the program will merge any two adjacent conserved blocks that separated by less than the user-defined minimum value set by `-f`. The default value is 3.

Format

```
>>> python seqfire.py -i inputfile.fa -f {number of residues}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -f 5
```

Profile Combination Option (-r)

The similarity and entropy profiles can then be combined to defend conserved blocks in two different ways, union or intersection. The user can specify this using `-r`. The choices are "True", which invokes the intersection method, or "False", which invokes the union method. The default option is "False" (union).

Format

```
>>> python seqfire.py -i inputfile.fa -r {True or False}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -r False
```

Special Options

Co-analysis (Indel Region & Conserved Block) Option (-e)

If the user wishes to obtain both conserved blocks and indel regions simultaniously, SeqFIRE allows this action using the `-e` option. This option has two alternative choices: True or False. Specifying `-e` True will add the indel matrix at the end of the alignment in NEXUS output. The default is 'False', which runs only the conserved block module.

Format

```
>>> python seqfire.py -i inputfile.fa -e {True or False}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -e True
```

Multiple Dataset Analysis Option (-m)

If the user wants to run SeqFIRE with multiple inputs, it is possible to pipeline SeqFIRE standalone into their process and setup a loop to manage the run. Alternatively, the user can use SeqFIREprep to combine all input files into a single large file. If this is done, the user can implement the -m option to analyze the batch (single large input) file. Choices for this option can be True (invokes batch mode) or False (single analysis mode). Default is False.

Format

```
>>> python seqfire.py -i inputfile.fa -m {True or False}
```

Example

```
>>> python seqfire.py -i inputfile.fa -a 2 -m True
```

SeqFIRE Quick Run

Indel Region Module

To run SeqFIRE quickly with default values use the commands below. If the input file is in the same folder as the SeqFIRE program use:

```
>>> python seqfire.py -i alignment.fa -a 1 -o 1
```

If the input file is not located in the same folder as SeqFIRE, you will have to include the path of the input file, *e.g.*:

```
>>> python seqfire.py -i c:\data\alignment.fa -a 1 -o 1
```

Conserved Block Module

To run SeqFIRE for identification of the conserved alignment blocks using default parameters, use the following command:

```
>>> python seqfire.py -i alignment.fa -a 2 -o 1
```

or

```
>>> python seqfire.py -i c:\data\alignment.fa -a 2 -o 1
```

Chapter 6
Error Messages

This chapter provides some suggestions and hints when users get either an **error message** or SeqFIRE fails to run or appears to run but produces no output.

Error Messages

Parameter Value out of Range

This error message will occur if user inputs invalid parameter values. For example, the user can put in values for inter-indel space between 1-10. Any value above 10 will result in the error message "parameter value out of range" and the program will terminate.

ERROR: PARAMETER VALUE OUT OF RANGE
The inter-indel space value is '12', which is out of range. It should be between 1 and 10.

Solution: Re-assign the parameter value in the range 1-10.

Input Conflict

This error message will appear when the user pastes an input file into the input box while simultaneously uploading a file. SeqFIRE will not run if more than one input is detected. Then, the program will warn the user with the following error message:

ERROR: INPUT CONFLICT
There is more than one input! If you want to upload an input file, please make sure that the input box is empty, and vice versa.

Solution: Make sure the input box is empty if uploading an input file. Or, if using the input box, make sure no file is specified next to the upload link.

No Input

The following error message will appear if no input is specified:

ERROR: NO INPUT
SeqFIRE couldn't run your task because there is no input data. Please make sure that you copy your input alignment in the input box or upload the input file, then run your task again.

Solution: User has to assign an input either in the input box or upload an input file.

Input Cannot Run

This error message will appear if SeqFIRE encounters a problem while running the analysis. SeqFIRE prints the error message with suggestions for some common mistakes.

ERROR: INPUT CANNOT RUN
SeqFIRE cannot run with your input file. Please try the following
(1) Check your input file for formatting errors such as non-standard symbols (e.g., gaps should be denoted by dash or dot).
(2) Make sure you have no completely blank sequences.
(3) Try running with less strict analysis parameters.
(4) Contact us if you think there might be a bug or if you need help.

Solution: A number of possible errors that might cause SeqFIRE to abort. The most common are

(1) Format of input, input MUST be an alignment in FastA format. The sequences MUST not contain any non-standard symbols (*e.g.* !, @, #, $, %, &, *, +, |, \, /, ~, etc.) or ambiguous amino acid symbols (B, J, O, U and Z). The gap characters MUST be designated by dash (-) and/or dot (.)

(2) Blank sequences that contain only gap character are not allowed.

(3) The maybe no output. This maybe because the analysis parameters are too strict. If you think the format is correct, try to run SeqFIRE with more lenient parameters. For example, assign 60% amino acid conservation threshold instead of the default (75%).

References

Ajawatanawong P, Atkinson GC, Watson-Haigh NS, MacKenzie B, Baldauf SL. (2012) SeqFIRE: a web application for automated extraction of indel regions and conserved blocks from protein multiple sequence alignments. *Nucleic Acids Res* **40**:W340-W347.

Castresana J. (2000) Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol Biol Evol* **17**:540-552.

Dayhoff MO, Schwartz RM, Orcutt BC. (1978). A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure* **5**:345–352.

Do CB, Mahabhashyam MSP, Brudno M, Batzoglou S. (2005) PROBCONS: Probabilistic Consistency-based Multiple Sequence Alignment. *Genome Res* **15**:330-340.

Edgar RC. (2004a) MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* **5**:113.

Edgar RC. (2004b) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res* **32**:1792-1797.

Henikoff S, Henikoff JG. (1992). Amino acid substitution matrices from protein blocks". *Proc Natl Acad Sci U S A* **89**:10915–10919.

Illergård K, Ardell DH, Elofsson A. (2009) Structure is three to ten times more conserved than sequence--a study of structural response in protein cores. *Proteins* **77**:499-508.

Katoh K, Standley DM. (2013) MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Mol Biol Evol* **30**:772-780)

Lassmann T, Sonnhammer EL. (2005) Kalign--an accurate and fast multiple sequence alignment algorithm. *BMC Bioinformatics* **6**:298.

Löytynoja A, Goldman N. (2008) Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science* **320**:1632-1635.

Löytynoja A, Goldman N. (2010) webPRANK: a phylogeny-aware multiple sequence aligner with interactive alignment browser. *BMC Bioinformatics* **11**:579.

Rost B. (1999) Twilight zone of protein sequence alignment. *Protein Eng* **12**:85-94.

Thompson JD, Koehl P, Ripp R, Poch O. (2005) BAliBASE 3.0: Latest developments of the Multiple Sequence Alignment Benchmark. *Proteins* **61**:127-136.

Waterhouse AM, Procter JB, Martin DMA, Clamp M, Barton GJ. (2009) Jalview version 2 - a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, **25**:1189-1191.

Weaver W, Shannon CE. (1963). The Mathematical Theory of Communication. Univ. of Illinois Press.

Wu M, Chatterji S, Eisen JA. (2012) Accounting for alignment uncertainty in phylogenomics. *PLoS ONE* **7**:e30288.